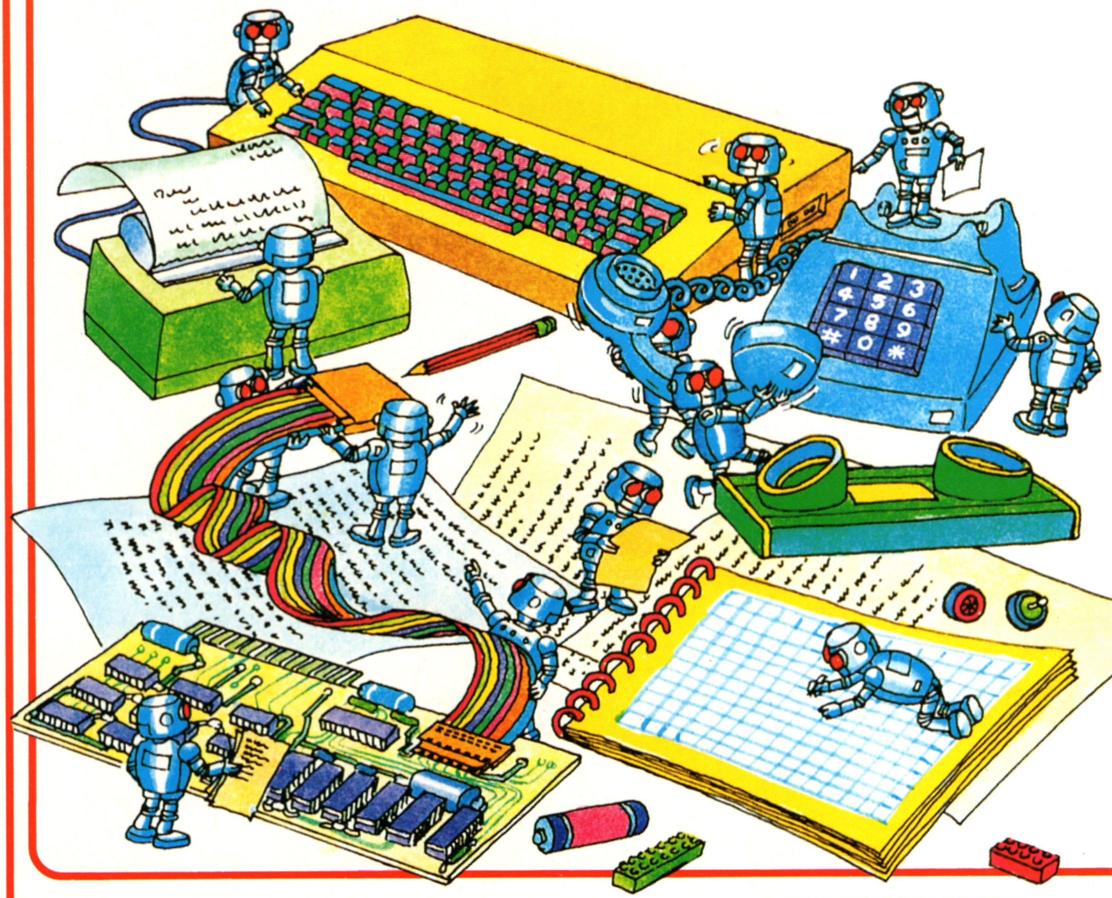
L. 2.500
Frs. 3,75

BEST-SELLER MONDIALE

11

LA GRANDE ENCICLOPEDIA DI INFORMATICA PER RAGAZZI

IN SOLI 30 FASCICOLI



Spedizione in Abb. Postale Gruppo II/70

GRUPPO EDITORIALE JACKSON

IN COLLABORAZIONE CON AMSTRAD

Direttore responsabile
Paolo Reina

Direttore di divisione:
Roberto Pancaldi

Autori:
Judy Tatchell,
Nick Cutler,

Lisa Watts,
Mike Wharton,

Tony Potter,
Ivor Guild,

Ian Graham,
Lynn Myring,

Helen Davies,
Mike Wharton,

Ian Graham,

Brian Reffin Smith,
Lisa Watts,

Bill Bennett,
Judy Tatchell,

Jenny Tyler,

Lee Howarth,
Judy Tatchell,

Gaby Waters,
Graham Round,

Nick Cutler,
Gaby Waters,

Brian Reffin Smith,

Judy Tatchell,
Lee Howarth,

Cherry Evans,
Lee Howarth

Revisione e adattamento:
Martino Sangiorgio

Coordinamento editoriale:
Renata Rossi

Progetto grafico:
Sergio Mazzali

Distribuzione:
SODIP - Milano

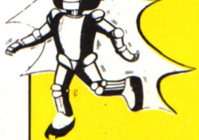
Stampa:
Vela - WEB - Vigano di Gaggiano (MI)

Direzione e Redazione:
Via Rosellini, 12 - Milano (20124) - Tel. 02/6880951 (5 linee)

Autorizzazione alla pubblicazione: richiesta
Spedizione in abbonamento postale Gruppo II/70
(autorizzazione della Direzione Provinciale delle PPTT di Milano)
Prezzo del fascicolo L. 2.500
I numeri arretrati saranno disponibili per 1 anno dal completamento dell'opera e potranno essere richiesti direttamente all'Editore a L. 3.000 (sovrapprezzo di L. 10.000 per spese d'imballo e spedizione).
I versamenti vanno indirizzati a:

Gruppo Editoriale Jackson S.p.A.
Via Rosellini, 12 - 20124 Milano

mediante emissione di assegno bancario oppure utilizzando il
C.C. Postale N. 11666203.
Non vengono effettuate spedizioni in contrassegno.



NEL PROSSIMO NUMERO:

- PROGRAMMA PER COMPORRE POESIE
- ALTRI USI DEL PERSONAL
- CHE COSA AGGIUNGERE AL VOSTRO COMPUTER
- GIOCO ADVENTURE: LE SUBROUTINES
- L'UFFICIO ELETTRONICO

Giochi con DATA

Un sistema semplice di immettere nel computer molte informazioni è la combinazione delle parole READ e DATA. Una riga DATA contiene un elenco di parole o numeri e READ dice al computer di memorizzare i dati in una o più variabili. A pagina 163 troverete però un metodo per immagazzinare i dati per mezzo di un vettore.

Leggere: DATA



Separate ogni elemento di DATA con una virgola.

```
10 FOR I=1 TO 7
20 READ X,X$
30 PRINT X;" ";X$
40 NEXT I
50 DATA 13,PESCI,77,RANE,91,LUMACHE
60 DATA 23,GATTI,62,CANI,2,TOPI,1,RATTO
```



Alcuni computer richiedono che le stringhe stiano tra virgolette.

Provate questo programma per vedere come funzionano READ e DATA. La parola READ è seguita da due variabili ed ogni volta che il ciclo si ripete il computer memorizza la successiva coppia di elementi dei DATA con le variabili X e X$.

Controllo del nome



Sostituite il punto interrogativo nella riga 50 con l'ultimo nome dei vostri DATA.

```
10 PRINT "IL NOME PREGO ";
20 INPUT N$
30 READ X$
40 IF X$=N$ THEN GOTO 70
*50 IF X$=" ? " THEN PRINT "IL VOSTRO NOM
E NON E' IN LISTA": STOP
60 GOTO 30
70 PRINT "O.K. IL VOSTRO NOME E' NELLA L
ISTA"
*80 DATA  ?
```

In questo programma il computer chiede il vostro nome e controlla che sia presente in una lista di nomi memorizzati come DATA. Provate a mettere i vostri propri dati alla riga 80 (potete aggiungere tutti i nomi che volete), quindi inserite l'ultimo nome nella riga 50 per fermare il computer dopo aver letto l'ultimo nome della lista.

Ripristinare i DATA

```
*10 FOR J= ? TO  ?
*20 FOR I= ? TO  ?
 30 READ N$
*40 IF LEFT$(N$,1)=CHR$(?) THEN PRINT N$
 50 NEXT I
 60 RESTORE
 70 NEXT J
 80 DATA VERA,SABINA,ZACCARIA,ORAZIO
 90 DATA BIANCA,DANIELA,TAMARA,PAOLO
 100 DATA GIORGIO,LUCA,ANGELA,MARCO
 110 DATA SAVERIO,CARMELO,LUCIA,JERRY
 120 DATA VITO,BICE,CLAUDIO
```

Ponete la variabile del ciclo J pari al numero delle lettere dell'alfabeto usando i numeri di codice dei caratteri.

Ponete la variabile del ciclo I pari al numero di elementi dei DATA.

Sapreste completare questo programma per disporre i nomi in ordine alfabetico? I nomi sono elencati nelle righe DATA; RESTORE dice al computer di ritornare all'inizio dell'elenco dei DATA ogni volta che si ripete il ciclo di J. Provate a completare le variabili e i numeri mancanti per far girare il programma.

Giochi per individuare gli errori.



1

```
10 FOR K=1 TO 6
20 READ N : PRINT N
30 NEXT K
40 DATA 02-56439,06-11678,015/77983
50 DATA 0931-87654,087/54678,088-54672
```

In questo programma alle righe DATA sono memorizzati alcuni numeri di telefono. Riuscite ad individuare gli errori nel programma e correggerli?

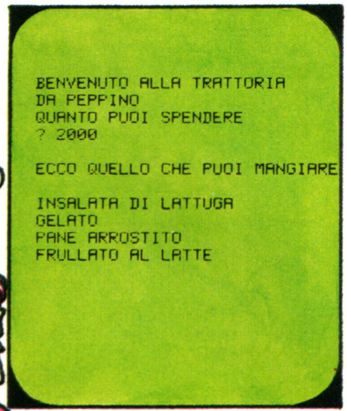
2

```
10 READ X
20 PRINT X
30 GOTO 10
40 DATA 1,461,892,66,1471,4462,1,3
50 DATA 53,80,241,90,371,825,33,13
```

Se non riuscite a trovare gli errori provate a farlo girare. Il computer dovrebbe stampare un messaggio di errore per dirvi cosa è sbagliato. Sapete ideare un modo semplice per risolvere questo problema?

Trattoria da Peppino



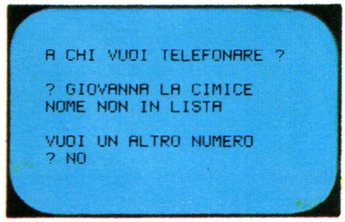


In alto a sinistra c'è il menu della trattoria "da Peppino". Utilizzando i prezzi e i componenti del menu come dati, provate a scrivere un programma che dica quanto potete mangiare data una certa somma di denaro, come è illustrato nello schermo a destra. Potete anche aggiungere qualche altra specialità al menu.

Guida telefonica

Troverete sotto alcune istruzioni per scrivere un programma per una guida telefonica. Gli schermi a destra mostrano come girano questi programmi. Provate a scrivere il programma.



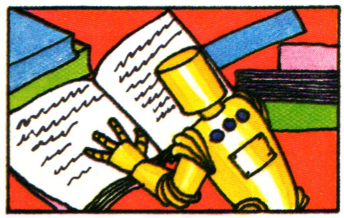




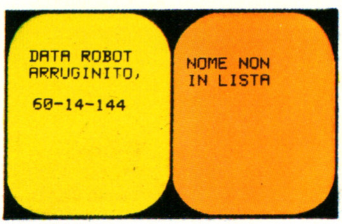
1. Compilate una lista di nomi di vostri amici e dei loro numeri di telefono con le istruzioni DATA, come mostrato sopra.



2. Usate PRINT per farvi chiedere dal computer a chi volete telefonare ed INPUT per la vostra risposta.



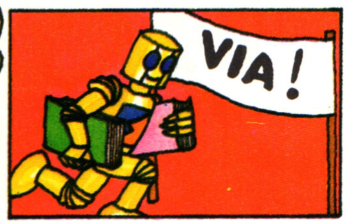
3. Per ricercare il nome usate READ all'interno di un ciclo. Usate variabili diverse per i nomi e i numeri.



4. Stampate il nome ed il numero (che dovrebbe essere il termine che segue il nome) oppure segnalate che il nome non è nella lista.



5. Fate chiedere al computer se volete un altro numero. Usate INPUT per la vostra risposta.



6. In funzione della risposta, ripristinate (RESTORE) i dati ritornando all'inizio o fermate il programma.

Usiamo i vettori

Un utile sistema per immagazzinare i dati è rappresentato dal vettore. Si può pensare al vettore come ad un insieme di variabili dove ogni dato è memorizzato in una casella numerata. Ciascun dato è chiamato elemento del vettore. Potete riferirvi ad un elemento usando il nome del vettore ed il suo numero di casella che è detto argomento.

Vettori numerici



Ecco un vettore numerico chiamato N che contiene sei elementi. Bisogna indicare al computer quanti elementi conterrà il vettore, in modo che venga riservato uno spazio sufficiente nella memoria del computer. Per fare questo usate la parola DIM seguita dal nome del vettore e dal numero degli elementi che contiene; questo viene detto dimensionamento del vettore.

```
*10 DIM ?
 20 FOR K=1 TO 6
 30 READ N(K)
 40 NEXT K
*50 DATA ?
```

```
*10 DIM ?
 20 FOR K=1 TO 6
 30 INPUT N(K)
 40-NEXT K
```

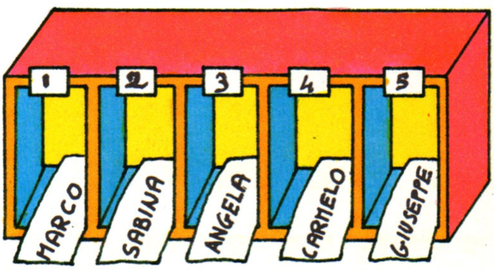
```
N( 1 )  E'   1066
N( 2 )  E'   1216
N( 3 )  E'   1485
N( 4 )  E'   1603
N( 5 )  E'   1665
N( 6 )  E'   1959
```

```
1216        1603
1603        1603
1216        1959
```

Per mettere i dati in un vettore potete usare un ciclo con READ/DATA. Provate a completare il programma sulla sinistra in modo che vengano immagazzinate in un vettore tutte le informazioni della figura in alto. Un altro modo per riempire un vettore è quello di usare istruzioni INPUT, come mostrato nel programma a destra.

Adesso provate a scrivere un programma per stampare sullo schermo i dati immagazzinati nel vettore. Usate PRINT e la variabile di ciclo come argomento del vettore. Sullo schermo a destra il computer stampa a caso gli elementi del vettore usando dei numeri casuali come argomento del vettore.

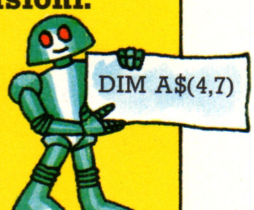
Vettori di stringa



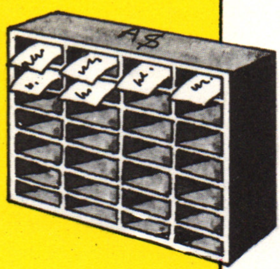
Questo è un vettore di stringa (N$). Esso contiene cinque nomi e così ha cinque elementi. I vettori di stringa vanno trattati allo stesso modo di quelli numerici. Cercate di scrivere un programma semplice per memorizzare i dati in questo vettore e poi stamparli sullo schermo.

Vettori a due dimensioni.

```
10 DIM A$(4,7)
20 FOR I=1 TO 4
30 FOR J=1 TO 7
40 READ A$(I,J)
50 NEXT J
60 NEXT I
70 DATA ........
```



I vettori possono anche avere due (o più) dimensioni. In tal caso vengono chiamati matrici. Il programma sopra esposto permette di inserire i valori all'interno della matrice mediante due cicli nidificati. Anche le matrici possono essere numeriche o di stringa. Potete pensare ad essa come ad un armadio con scomparti sia in orizzontale che in verticale.



Il calendario

```
* 10 ?
  20 FOR K=1 TO 12
* 30 ?
  40 NEXT K
  50 PRINT "NUMERO DEL MESE ";
  60 INPUT N
* 70 PRINT M$(?);" HA ";
* 80 PRINT D(?);" GIORNI"
* 90 DATA ?
* 100 DATA ?
* 110 DATA ?
```

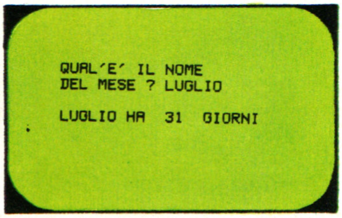
Dimensiona il vettore (riga 10)

Memorizza i dati letti (READ) nei due vettori M$ e D. (riga 30)

Completate gli argomenti mancanti per stampare i dati esatti dei vettori. (righe 70–80)

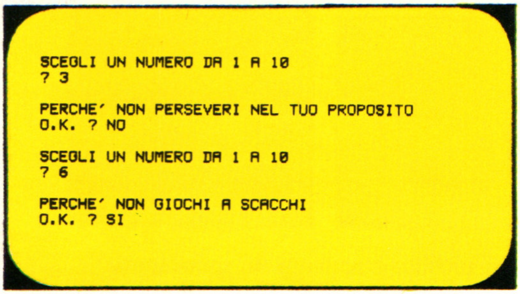
Ponete il nome di ogni mese seguito dal suo numero di giorni nelle istruzioni DATA di queste righe. (righe 90–110)

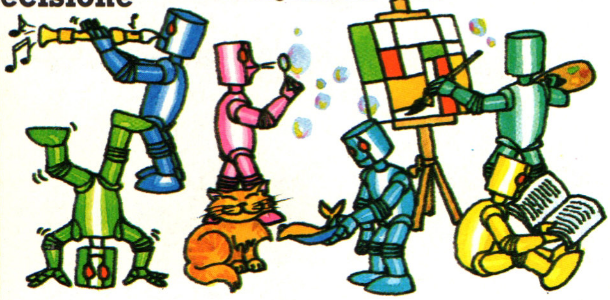
Sapreste completare questo programma in modo che battendo il numero di un mese, il computer stampi il nome del mese ed il numero di giorni di cui è composto? Vi sono alcuni suggerimenti di aiuto a fianco del programma.



Modificate il programma in modo che il computer chieda il nome di un mese e poi ve ne dica il numero dei giorni. Usate un ciclo e IF/THEN per cercare in M$ il nome del mese. Servitevi della variabile del ciclo come argomento per scegliere l'elemento esatto immagazzinato in D.

Programma per prendere una decisione

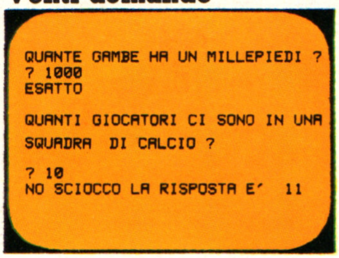




Ecco un'idea per un programma che potrebbe essere molto utile quando non sapete decidere sul da farsi. Per far girare il programma scegliete un numero ed il computer farà apparire un suggerimento sullo schermo.

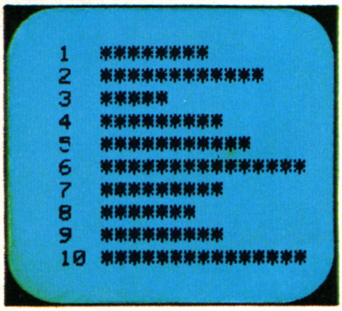
Per scrivere il programma avete bisogno di un vettore di stringa (I$) contenente dieci idee. Inserite (INPUT) un numero con una variabile (N) e fate prendere al computer uno degli elementi del vettore usando N come argomento di I$.

Venti domande



Questo schermo illustra il risultato di un gioco a quiz. Per scrivere il programma, componete 20 domande e mettetele in un vettore di stringa. Mettete poi le risposte in un altro vettore. Gli argomenti dei due vettori dovrebbero collegare le domande alle relative risposte.

Tabella con numeri casuali






```
10 LET N=0
20 DIM A(10)
30 FOR K=1 TO 10
40 LET A(K)=0
50 NEXT K
60 LET R=INT(RND(1)*10+1)
70 LET A(R)=A(R)+1
80 LET N=N+1
90 IF N<100 THEN GOTO 60
```

Questo programma usa un vettore per immagazzinare i dati da mettere in tabella. Il computer sceglie a caso 100 numeri compresi tra 1 e 10. Ogni elemento del vettore (A) conta quante volte ciascun numero viene scelto. Cercate di completare il programma in modo che il computer mostri una tabella come quella sopra dove viene stampato un asterisco ogni volta che il numero viene selezionato. Per fare questo scrivete un ciclo che si ripete 10 volte con un altro ciclo all'interno di esso. Il ciclo interno dovrebbe stampare una riga della tabella per volta.

Oroscopo

Questo programma stampa un oroscopo quando qualcuno immette la sua data di nascita. Mostra come il computer usa le matrici per immagazzinare liste di informazioni in un ordine particolare, come segni zodiacali, mesi e oroscopi.

```
10 PRINT "OROSCOPO"
20 DATA 0,31,59,90,120,151,181,212,243,273,304,334
30 DIM M(12)

40 FOR I=1 TO 12:READ M(I):NEXT I
50 DATA ACQUARIO,20,PESCI,50,ARIETE,80
60 DATA TORO,111,GEMELLI,141,CANCRO,172
70 DATA LEONE,203,VERGINE,234,BILANCIA,265
80 DATA SCORPIONE,296,SAGITTARIO,326
90 DATA CAPRICORNO,355

100 DATA "INCONTRERAI UNO STRANIERO CUPO E MISTERIOSO"
110 DATA "SARAI RICCO E FELICE SCRIVENDO PROGRAMMI"
120 DATA "CONOSCERAI IL SENSO DELLA VITA"
130 DATA "SARAI MOLTO FELICE VIVENDO AD HOLLYWOOD"
140 DATA "T'INNAMORERAI DI UN COMPUTER VERDE"
150 DATA "SARAI PIU' FELICE LONTANO DA QUI"
160 DATA "IMPARA AD AMARE RANE E ROSPI"
170 DATA "NON CORRERE DIETRO ALL'AUTOBUS SE PIOVE"
180 DATA "I TUOI FIGLI NON CREDERANNO ALL'OROSCOPO"
190 DATA "DI' SEMPRE LA VERITA' AL GIOVEDI'"
200 DATA "LA TUA FACCIA SARA' LA TUA FORTUNA"
210 DATA "NON CURARTI DEGLI OROSCOPI"

220 DIM S$(12),S(12)
230 FOR I=1 TO 12
240 READ S$(I),S(I)
250 NEXT I

260 DIM D$(12)
270 FOR I=1 TO 12
280 READ D$(I)
290 NEXT I

300 PRINT:PRINT:PRINT
310 PRINT "IMMETTI IL NUMERO DEL TUO"
320 PRINT "MESE DI NASCITA"
330 PRINT "(GEN=1,FEB=2,ECC.)"
340 INPUT M
350 IF M<1 OR M>12 THEN PRINT "MESE INESISTENTE":GOTO 300
360 PRINT
370 PRINT "IMMETTI IL GIORNO"
380 INPUT D
390 IF D<1 OR D>31 THEN PRINT "GIORNO INESISTENTE":GOTO 360
400 X=M(M)+D
410 A=0

420 FOR I=1 TO 11
430 IF (X>=S(I)) AND (X<S(I+1)) THEN A=I

440 NEXT I
450 IF A=0 THEN A=12
460 CLS
470 PRINT:PRINT:PRINT
480 PRINT "IL TUO SEGNO E': ";S$(A)
490 PRINT
500 PRINT D$(A)
510 PRINT:PRINT
520 GOTO 300
```

Linea di DATA contenente il numero del giorno di inizio di ogni mese, ad es. Aprile inizia il 90° giorno dell'anno.

Segni zodiacali e numero del giorno in cui iniziano.

Oroscopi per ogni segno zodiacale.

Potete creare nuovi oroscopi o allungarli. Leggete quelli delle riviste per prendere spunti.

Prepara la matrice per i segni (S$) ed i loro numeri (S); DIM dice al computer quanti dati saranno posti nella matrice. READ trasferisce ogni elemento DATA nella matrice e lo numera.

Prepara la matrice per gli oroscopi.

Riuscite a aggiungere qualche istruzione per dire al computer cosa fare se il numero del mese o del giorno è immesso in lettere invece che in cifre?

Il computer controlla che abbiate immesso un numero fra 1 e 12 per il mese e fra 1 e 31 per il giorno.

Il computer calcola il numero del giorno del vostro compleanno e controllando il numero del giorno iniziale del mese nella matrice M e aggiungendovi il numero che avete immesso con D.

Controlla il vostro numero (X) e ricerca il vostro segno (S).

Stampa il segno zodiacale e l'oroscopo corrispondente.

Telesoftware

Se possedete un home-computer, avete bisogno di programmi per farlo funzionare. Il telesoftware è un modo nuovo e molto conveniente per avere programmi per il computer, tramite il videotex. Il software viene caricato direttamente nel vostro computer: esso può giungervi sotto forma di trasmissione di segnali del teletext o del videodata, entrambi tramite TV via cavo interattivo o mediante linea telefonica. Utilizzare il telesoftware è molto piú semplice che digitare lunghi programmi riportati su riviste e può essere piú economico che comperare cassette registrate.

Come ottenere il telesoftware

I programmi del telesoftware sono scritti sotto forma di pagine di testo, memorizzate nel database delle società di videotex e trasmesse esattamente come ogni altro tipo di informazione del videotex stesso. Il telesoftware del teletext è trasmesso via etere, il telesoftware del videodata viene inviato mediante linea telefonica o TV via cavo. Per ricevere il videotex avrete bisogno di alcuni dispositivi aggiuntivi, come un adattatore o un modem. Potrebbe anche essere necessario un software specifico che consenta al vostro computer di comprendere la codifica del videotex.
I servizi di telesoftware spesso forniscono molto di piú di semplici programmi: ad esempio pagine di novità sui computer e sul software, riviste e suggerimenti. Potreste dover sostenere delle spese per alcuni programmi e per il servizio di telesoftware stesso, se quest'ultimo vi dovesse pervenire come videodata a pagamento, via TV o via telefono. Il teletext è generalmente gratuito.

Il telesoftware del teletext

Il telesoftware del teletext ha meno programmi disponibili, e piú brevi, di quello del videodata. Occorre uno speciale adattatore/ricevitore, per ricevere i segnali del teletext. Questo perché, se state utilizzando il vostro apparecchio TV come schermo per l'home-computer, esso sarà sintonizzato sul canale utilizzato dal computer stesso e quindi non potrà ricevere contemporaneamente il canale del teletext.
L'adattatore provvede anche a decodificare i segnali del teletext, in modo che non avrete bisogno di un televisore speciale dedicato appositamente ad esso o di un software aggiuntivo. Un ulteriore motivo per avere un adattatore/ricevitore separato è che il TV non può immettere le informazioni del teletext nel computer.

Software via cavo

Se il telesoftware fa parte di un servizio interattivo via cavo TV, si troverà probabilmente su un canale a pagamento destinato al videodata. Dovete pagare per vedere il canale, cosí come per il software che volete. Avrete bisogno di uno speciale adattatore inserito nel vostro TV in modo che questo possa ricevere i segnali ed inviarli al vostro computer, cosí come avrete bisogno di un opportuno software per consentire al vostro computer di decodificare questi segnali. Un giorno potrà anche essere possibile usare il telesoftware tramite il videodata pur senza possedere un computer. I computers centrali dei videodata potrebbero eseguire i programmi per voi, visualizzarne i risultati in forma di pagine ordinarie da controllare con la normale tastiera del teletext.





Software via telefono

Per ricevere telesoftware da un servizio di videodata mediante telefono dovrete essere attrezzati per ricevere il videodata come descritto alle pagine 126-127. Il videodata telefonico sarà probabilmente fornito da un gruppo chiuso di utenti che hanno affittato spazio nel database del videodata. Una volta che vi siete iscritti per ricevere il telesoftware, sarete anche autorizzati a ricevere gli altri servizi in videodata. Il disegno seguente mostra un home-computer che riceve un programma di giochi attraverso il telefono, usando un modem ad accoppiatore acustico.

Caricamento di telesoftware

Immettere telesoftware nel vostro computer è detto trasferimento dal sistema centrale (download). Voi scegliete, attraverso dei menu, i programmi che desiderate; essi sono solitamente raggruppati per argomento, giochi, affari, istruzione e cosí via, ma soprattutto a seconda del tipo di computer. Ciò perché i programmi scritti per un certo computer non funzionano su uno di tipo diverso. I programmi per computer sono scritti in linguaggi speciali, solitamente Basic o linguaggio macchina per home-computers, ma sfortunatamente differenti tipi di computers utilizzano differenti versioni di Basic e differenti linguaggi macchina, per cui sarete in grado di trasferire dal sistema centrale solamente il software scritto appositamente per il vostro tipo di computer. Mentre è in trasferimento, un programma può apparire sul vostro schermo TV come una serie di righe formate da lettere o simboli senza senso. Ciò perché i programmi vengono trasmessi in una speciale codifica compattata, che è piú breve e quindi piú veloce da inviare che non il linguaggio ordinario per computer. Una volta caricato completamente, il programma può essere avviato, memorizzato su nastro o disco per impieghi futuri, visualizzato sullo schermo o stampato se desiderate vederlo scritto.

TV utilizzato come schermo display per il telesoftware

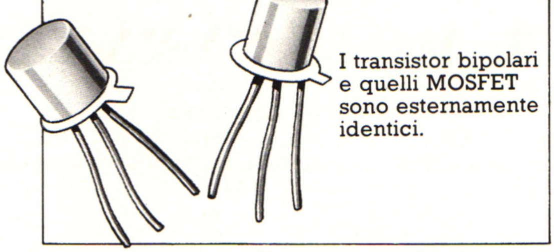


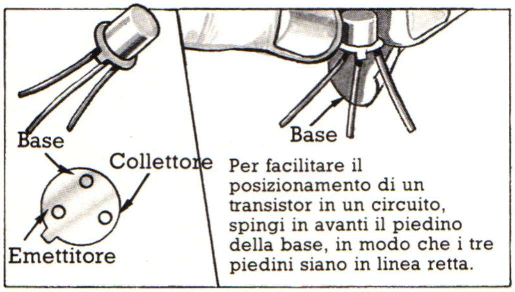
Telefono e modem ad accoppiatore acustico

Suggerimenti per costruire circuiti

Le due pagine seguenti forniscono informazioni sulle componenti elettroniche e sulla saldatura e ti aiuteranno nella costruzione del circuito logico delle pagine 132-133.

Transistor

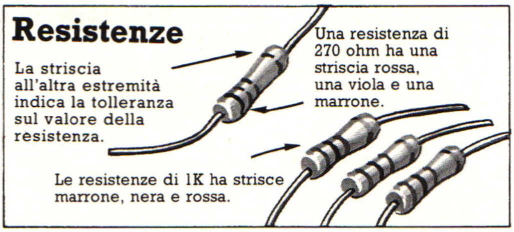




I transistor necessari per il circuito logico sono chiamati bipolari, o transistor a giunzione. Sono leggermente diversi da quelli chiamati FET, (dall'inglese a "transistor a effetto di campo"), anche se fanno passare o bloccano una corrente nello stesso modo.

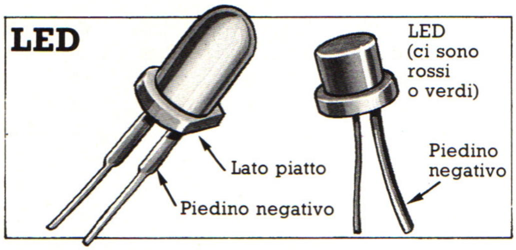
I tre piedini di un transistor bipolare sono chiamati emettitore, base e collettore. Una corrente può scorrere nel transistor solo fra l'emettitore e il collettore, purchè venga inviata una corrente alla base. La figura qui sopra mostra, ad esempio, come riconoscere i vari piedini sul transistor BC107.

Resistenze



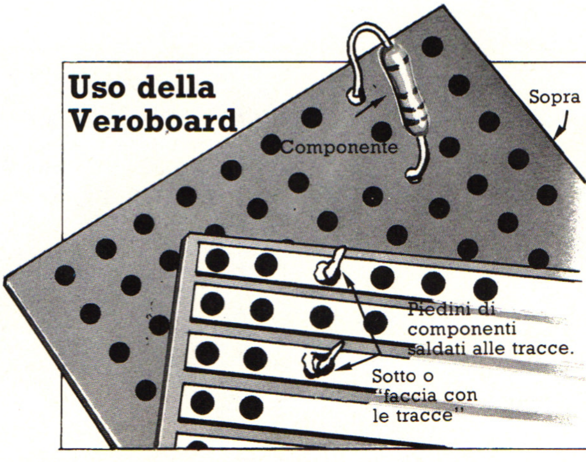
Le resistenze riducono la corrente proveniente dalla pila a un livello compatibile con le altre componenti del circuito. La loro forza (cioè la quantità di cui riducono la corrente) viene misurata in ohm * ed è mostrata dalle tre strisce colorate a un'estremità della resistenza. Quando si parla di componenti, K sta per 1000, quindi una resistenza di 1K ha una forza di 1000 ohm.

LED



LED sta per *Light Emitting Diode* (diodo emettitore di luce). Un diodo è una componente attraverso la quale la corrente può scorrere solo in una direzione e i LED sono diodi che si illuminano quando vi passa corrente. I LED hanno un piedino negativo e uno positivo ed è importante che siano disposti nel circuito in modo corretto. Il piedino negativo di alcuni LED è piú spesso; il corpo di altri LED ha un lato schiacciato e il piedino negativo è quello che gli è piú vicino.

Uso della Veroboard

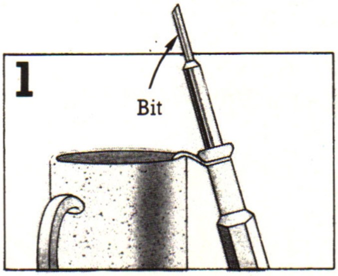


La Veroboard è stata appositamente progettata per costruire semplici circuiti elettronici. Ha file di fori, collegati fra loro da tracce di rame sul retro. I piedini delle componenti e i fili provenienti da una pila vengono inseriti nei fori e saldati al rame, cosí che la corrente può scorrere fra le componenti lungo le tracce. Le dimensioni di una Veroboard sono date dal numero di tracce per quello dei fori su ciascuna traccia (per es.: 10 tracce x 24 fori).

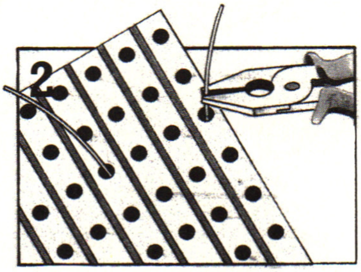
*Il simbolo per ohm è Ω.

Come si salda

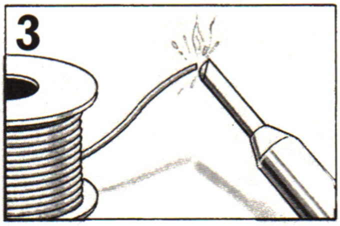




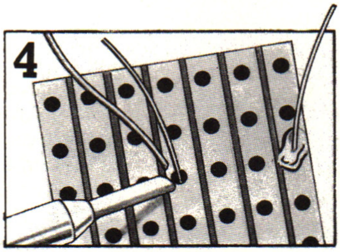
Attacca alla corrente il saldatore. Mentre si riscalda appoggialo in modo che la punta non tocchi nulla.



Per saldare una componente su una Veroboard, trova i fori giusti, fai passare i piedini e piegali leggermente con una pinza.



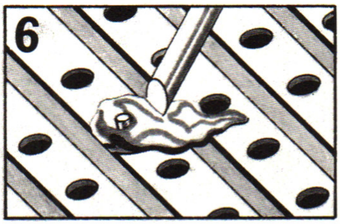
Con la punta calda del saldatore, tocca l'estremità dello stagno in modo che se ne sciolga una goccia e rimanga attaccata alla punta.



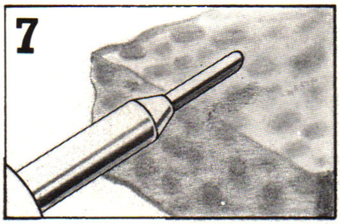
Tocca il piedino della componente con la punta del saldatore e l'estremità dello stagno per un secondo, finché una goccia di stagno si attacca alla traccia.



Fai raffreddare la saldatura, poi inclina la scheda allontanandola da te e taglia i piedini vicino alla saldatura usando un tagliafili.



E' molto importante togliere lo stagno eventualmente caduto nei solchi fra le tracce. Passa la punta calda lungo la traccia.



Dopo ogni saldatura, pulisci la punta sulla spugna umida e *ricorda di staccare il saldatore quando hai finito.*

Dissaldare

Per togliere, o dissaldare, una componente, inserisci la punta di una matita fra i piedini della componente sulla faccia superiore della Veroboard. Qualcuno, deve aiutarti ad inclinare la Veroboard e a tenere la matita, tirando via la componente mentre fai sciogliere le saldature sulle tracce con il saldatore.

Come stagnare un filo

Se usi filo a treccia, conviene rivestirne (stagnarne) le estremità con uno strato di lega per saldature, in modo che sia pú facile farle passare per i fori sulla scheda.

Con un paio di pinze, togli circa un centimetro del rivestimento plastico del filo e attorciglia i vari fili che lo compongono.

Metti qualcosa di pesante sul filo per tenerlo fermo. Tocca qualche volta i fili con la lega e la punta, cosí da rivestirli.

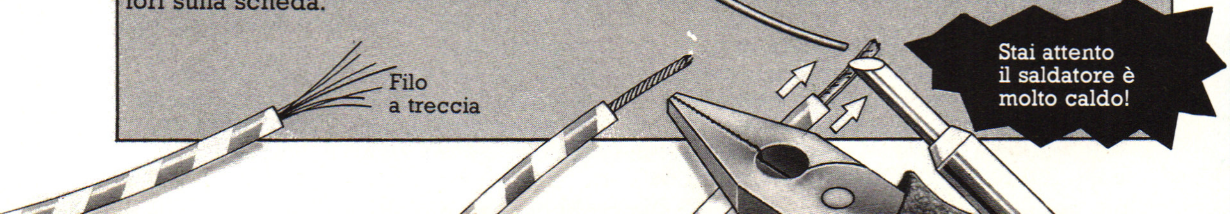
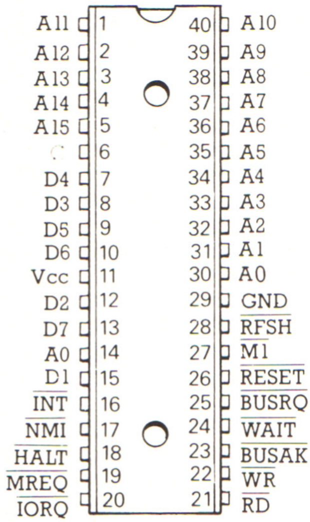


Tavola dei piedini di un microprocessore

I piedini dell'involucro di un microprocessore trasportano elettricità e indirizzi, dati e segnali di controllo dentro e fuori dal chip. La figura a destra è un diagramma, o la tavola dei piedini, di uno dei microprocessori piú diffusi, lo Z80. (Z sta per Zilog, il nome del costruttore). I piedini sono numerati da 1 a 40, con il numero 1 a sinistra dell'intaglio nella parte superiore dell'involucro. Le etichette sui piedini indicano il segnale che trasportano; qui sotto spieghiamo cosa significano e qual'è il compito di ciascun segnale. Su molte etichette vediamo una riga che indica che il piedino è attivo quando è basso.



A0-A15 Questi piedini trasportano i codici di indirizzo a 16 bit che escono dal microprocessore. A sta per *address* (indirizzo) e il numero si riferisce alla posizione del bit nel codice.
D0-D7 Questi piedini trasportano i codici a otto bit dei dati al e dal microprocessore. D sta per dato.
Ø Questo piedino trasporta il segnale del clock (rappresentato dalla lettera greca fi) al microprocessore.
Vcc Questo piedino è per l'alimentatore. E' collegato a +5 volt.
INT Sta per interruzione. Questo segnale interrompe le operazioni del microprocessore per farlo reagire a un'emergenza esterna, per esempio a un surriscaldamento della macchina.
NMI Sta per *non-maskable interrupt* (interruzione non mascherabile). E' un secondo segnale di interruzione che prevale sul precedente. Si chiama non mascherabile perché non c'è niente che possa annullarlo, o mascherarlo.
HALT E' un segnale che il microprocessore invia agli altri chip per comunicare di aver temporaneamente sospeso le proprie operazioni.
MREQ Sta per *memory request* (richesta di memoria). E' un segnale di controllo che il microprocessore invia per informare i chip di memoria che sul bus degli indirizzi c'è un indirizzo.
IORQ Sta per *input/output request* (richiesta di input/output. E' simile al segnale di richiesta di memoria solo che è rivolto ai dispositivi di input e di output per informarli che sul bus degli indirizzi c'è un indirizzo.
RD *Read* (lettura). E' il segnale di controllo che indica che ci sono dati da leggere da una posizione di memoria o da un dispositivo di input/output.
WR *Write* (scrittura). Il segnale di controllo che indica che vanno scritti dati in una posizione di memoria o in un dispositivo di input/output.
BUSAK Questo segnale e quello BUSRQ vengono utilizzati quando il microprocessore condivide i bus degli indirizzi e dei dati con altri microprocessori. BUSAK sta per *bus ackowledge* (riconoscimento del bus) ed è il segnale che il microprocessore invia a un altro pocessore per informarlo che i bus degli indirizzi e dei dati sono liberi.
WAIT (Aspetta) E' il segnale inviato al microprocessore per farlo aspettare qualche ciclo di clock perché succeda qualcosa all'esterno. Per esempio, alcuni tipi di chip di ROM lavorano piú lentamente della maggior parte dei chip di memoria, quindi il microprocessore deve aspettare piú a lungo che quei chip prendano. Il microprocessore inizia ad aspettare quando il voltaggio del piedino si abbassa e aspetta finché non risale.
BUSRQ Sta per *bus request* (richiesta di bus). E' il segnale inviato da un altro microprocessore che condivide i bus per far saper a questo microprocessore che vuole utilizzarli.
RESET (Azzeramento) E' il piedino che trasporta il segnale di azzeramento che riporta a 0 il contatore di programma all'accensione del processore (vedi pagina 118).
M1 E' il segnale che il microprocessore invia agli altri chip per informarli che sta prendendo un'istruzione dalla memoria.
RFSH Sta per *memory refresh* (ripristino della memoria). E' il segnale che il microprocessore invia ai chip della RAM dinamica per conservare intatti i dati che contengono.
GND Sta per *ground* (terra). Questo piedino è l'uscita dal microprocessore all'alimentatore. Viene chiamato terra perché è collegato a 0 volt.

Termini del chip

ACIA Un chip di interfaccia comune che gestisce le conversioni seriale/parallelo. ACIA sta per *Asynchronous Communication Interface Adaptor* (interfaccia di adattamento di comunicazioni asincrone)

Chip di RAM dinamica Un chip di RAM che, per non perdere i propri dati, deve essere costantemente ripristinato (*refresh*) con segnali elettrici.

Chip di RAM statica Un tipo di chip di RAM che non richiede segnali particolari per poter conservare i dati che contiene al momento dell'attivazione (vedi anche chip di RAM dinamica).

CMOS Un tipo di chip che contine MOSFET (vedi sotto) sia a canale n che a canale p. Richiede pochissima elettricità. CMOS sta per *Complementary Metal Oxide Semiconductor* (semiconduttore complementare a ossido di metallo).

EAPROM o EAROM Sta per *Electrically Alterable (Programmable) ROM* (Rom modificabile (programmabile) elettricamente). E' simile a un chip di EPROM, descritto sotto, ma i suoi programmi vengono cancellati inviando una corrente a determinati piedini. Chiamato anche EEPROM o EEROM (*Electrically Erasable*, cancellabile elettricamente).

EPROM Un chip di ROM simile a un PROM (vedi sotto), però è possibile cancellarne i programmi e inserirne di nuovi con un procedimento che implica l'investimento del chip con un fascio di luce ultravioletta. A questo scopo le EPROM hanno sempre un foro sopra l'involucro.

Flip-flop Un circuito elettrico ottenuto usando diversi transistor che possono assumere uno di due stati, per rappresentare 0 e 1. I flip-flop vengono utilizzati nei chip di memoria per formare celle e nei microprocessori per formare registri.

LSI Sta per *Large Scale Integration* (integrazione su grande scala) Di solito la scala di integrazione si riferisce al numero di componenti su un chip. Nei chip LSI ce ne sono da 100 a 10.000.

MOS Sta per *Metal Oxide Semiconductor* (semiconduttore a ossido metallico). Descrive la tecnologia utilizzata per produrre la maggior parte dei chip, che utilizzano il metallo come conduttore e il diossido di silicio come isolante.

MOSFET Sta per *Metal Oxide Semiconductor Field Effect Transistor* (transistor a ossido di metallo a effetto di campo). I tipi di MOSFET sono due: a canale n e a canale p, ottenuto con due isole di silicio di tipo p in uno strato di tipo n.

MSI Sta per *Medium Scale Integration* (integrazione di media scala). I chip MSI hanno fra 10 e 100 componenti.

NAND Un tipo di porta logica che esegue l'operazione opposta di una porta AND. NAND sta per "non AND".

NMOS Un tipo di chip che contiene solo MOSFET (vedi sopra) a canale n. I chip NMOS sono molto rapidi.

PMOS Un tipo di chip che contiene solo MOSFET (vedi sopra) a canale p. I chip PMOS possono utililizzare una notevole quantità di corrente.

PROM Un tipo speciale di chip di ROM nel quale è possibile scrivere programmi subito dopo la fabbricazione fondendo minuscoli fusibili incorporati nei circuiti. PROM sta per ROM programmabile.

SSI Sta per *Small Scale Integration* (integrazione su piccola scala). I chip SSI hanno meno di 10 componenti.

TTL Sta per *Transistor Transistor Logic* e descrive i chip con circuiti logici basati su transistor bipolari (non MOSFET).

UART Un diffuso chip di interfaccia che gestisce conversioni seriale/parallelo. UART sta per *Universal Asynchronous Receiver/Transmitter* (trasmettitore/ricevitore asincrono universale)

ULA Sta per *Uncommitted Logic Array* (matrice logica non predisposta). E' un chip con porte logiche che un costruttore può collegare in modi diversi secondo l'utilizzo.

VIA Un tipo di chip di interfaccia in grado di gestire tutti i tipi di conversione di segnali (per es.: analogico/digitale, seriale/parallelo). VIA sta per *Versatile Interface Adaptor* (adattatore di interfaccia versatile).

VLSI Sta per *Very Large Scale Integration* (integrazione su scala grandissima). I chip VLSI hanno piú di 10.000 componenti.

Tavola verità per AND

INPUT 1	INPUT 2	OUTPUT
0	0	0
1	0	0
0	1	0
1	1	1

Tavola verità per OR

INPUT 1	INPUT 2	OUTPUT
0	0	0
1	0	1
0	1	1
1	1	1

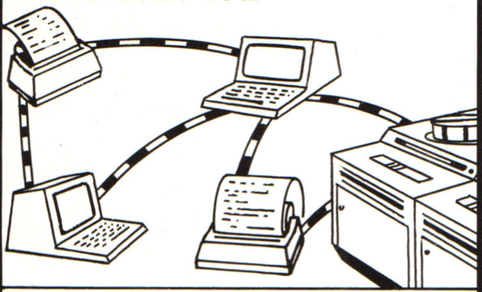
Tavola verità per XOR

INPUT 1	INPUT 2	OUTPUT
0	0	0
1	0	1
0	1	1
1	1	0

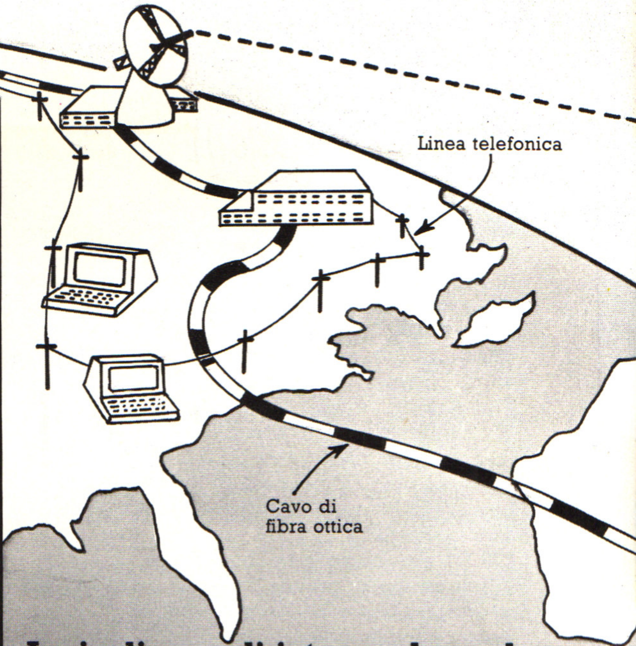
Reti di computer

Un personal computer può essere collegato a un altro in qualsiasi parte del mondo, purché ci siano le connessioni necessarie e il modo di trasmettere chiaramente i segnali. Si possono utilizzare i mezzi di comunicazione esistenti, come le linee telefoniche e i satelliti. Di solito per capirsi i computer hanno bisogno di programmi speciali, poiché possono usare linguaggi o dialetti diversi, oppure lavorare a velocità diverse. I computer vengono collegati fra loro per scambiarsi informazioni o programmi. Tutto ciò che è nella memoria di un computer può essere copiato in quella di un altro.

L'uso delle reti

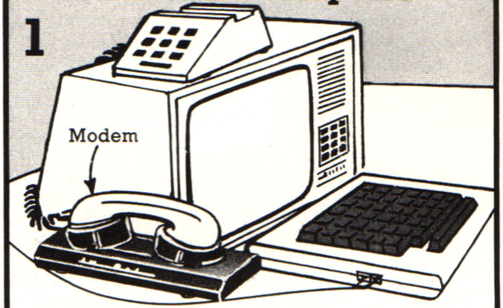


I computer possono essere collegati fra loro in reti, di solito tramite il telefono. Affinché l'altro computer riceva i tuoi messaggi, devi avere una parola d'ordine, dopo di che puoi collegarti con qualsiasi apparecchiatura; per esempio, piú personal computer potrebbero utilizzare la stessa stampante.



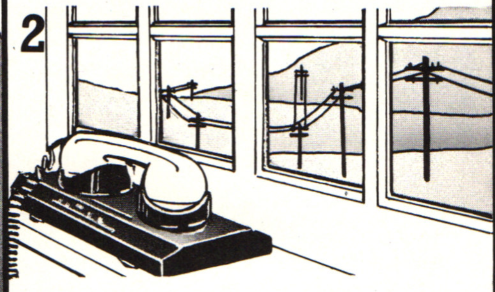
Telefonate fra computer

1



I computer possono essere collegati fra loro tramite telefono e un'apparecchio chiamato modem*, che trasforma i segnali in linguaggio macchina in segnali elettronici trasferibili su linee telefoniche.

2



Chi riceve i messaggi deve avere un altro modem collegato al proprio computer per ritrasformare i segnali elettronici in linguaggio macchina.

Invio di segnali intorno al mondo

I segnali dei computer possono essere inviati, sotto forma di onde radio, a un satellite, sul quale rimbalzano per arrivare in un determinato punto della terra. I satelliti di questo tipo trasmettono in tutto il mondo anche le chiamate telefoniche e i programmi TV.

Sono in corso di sviluppo nuovi metodi per la trasmissione dei segnali elettrici alla velocità della luce che utilizzano le fibre ottiche. I segnali in linguaggio macchina vengono trasformati in impulsi luminosi e cavi in fibra ottica li trasportano sopra la terra o sotto i mari in qualsiasi punto del globo.

* Modem sta per modulatore/demodulatore.

Nell'interno del programma

Ora che avete un'idea generale su come si presenterà il programma, potete cominciare a pensare a ciascuna parte più in dettaglio. Avete già visto in precedenza come funziona la fase di inizializzazione. Le pagine seguenti descrivono il funzionamento delle altre parti principali del programma.

Descrizione e risposta

Ad ogni mossa, il computer deve dire al giocatore dove questi si trovi, e in quali direzioni possa andare. Deve anche far sapere al giocatore quello che è successo in conseguenza delle sue ultime istruzioni. Questa è la fase della descrizione e risposta, e funziona più o meno così. Provate ad identificare ciascuna parte sul listato finale del programma.

```
90 PRINT "TITOLO DEL GIOCO"
100 PRINT "--------------"
110 PRINT "LA TUA LOCAZIONE"
120 PRINT D$(RM)
```

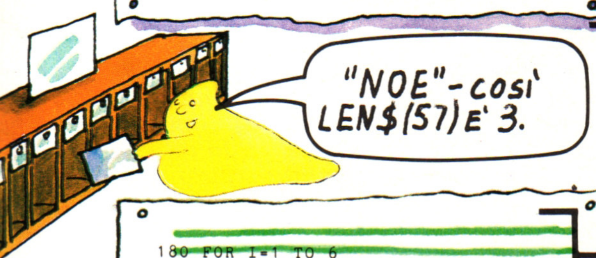
RM è il numero della locazione in cui si trova il giocatore. Dovrete ricordarvi di dare un valore iniziale a questa variabile, nella fase di inizializzazione. (Per "La Casa Stregata", il valore iniziale per RM è 57, vedere la linea 2090).

Il computer cerca in D$ (la matrice con tutte le descrizioni) e stampa ciò che trova nella casella numero RM.




```
130 PRINT "USCITE:";
140 FOR I=1 TO LEN (R$(RM))
150 PRINT MID$(R$(RM),I,1);",";
160 NEXT I
```

Questa routine esamina la lunghezza della stringa di caratteri nella casella numero RM della matrice dei percorsi, R$. Il computer quindi cicla per un numero di volte pari a tale valore, stampando uno per volta i caratteri del contenuto di R$(RM), ponendo tra ciascuno una virgola e lo spazio bianco.



Le istruzioni 180-200 costituiscono un ciclo per vedere se nella locazione c'è un oggetto il cui indicatore è nullo (ovvero, se l'oggetto è visibile), e stamparne il nome se c'è.

```
180 FOR I=1 TO 6
190 IF L(I)=RM AND F(I)=0 THEN PRINT
    "QUI PUOI TROVARE";O$(I)
200 NEXT I

220 PRINT M$

225 M$="COSA ?"
```

M$ è una variabile per contenere eventuali messaggi del computer al giocatore, in risposta a istruzioni date nel corso della mossa precedente. Cercate M$ nel listato del programma per vedere come diverse risposte vengano inserite in M$ in funzione di ciò che il giocatore aveva inserito da tastiera.

All'inizio di ciascuna mossa il contenuto di M$ è inizializzato a: "COSA?" per cui se ciò non viene sostituito da altri messaggi, il computer si limita a stampare "COSA?", richiedendo la successiva azione del giocatore.



La sezione d'inserimento

Un'importante caratteristica dei giochi di avventura consiste nel modo in cui il computer risponde ad istruzioni inserite da tastiera. "La Casa Stregata", come molti altri giochi, limita il giocatore all'uso di frasi di due parole, oltre all'uso di speciali istruzioni di una parola sola, come "AIUTO". La prossima sezione del programma dovrà chiedere istruzioni al giocatore, e quindi comunicare al computer cosa fare.

Per cominciare, il computer dovrà dividere la frase inserita dal giocatore in due parole, che potrà poi usare per un confronto con quelle che ha in memoria. La routine di divisione in parole utilizzata nella "Casa Stregata" funziona scandendo la frase inserita dal giocatore, finché non viene trovato uno spazio bianco tra le lettere. Tale routine viene listata qui, con l'aggiunta di poche linee per darvi modo di inserirla anche da sola, per vedere come funziona.

```
100 CLS
110 PRINT "BATTERE QUALCHE COSA"
120 INPUT Q$
130 V$=""
140 W$=""
```

Richiedono l'istruzione al giocatore, la ricevono e la scrivono in Q$. Vengono poi predisposte due nuove variabili stringa: V$ e W$.

```
150 FOR I=1 TO LEN(Q$)
```

Serve a vedere di quanti caratteri è composta Q$ e ad iniziare un ciclo per tale numero di volte.

```
160 IF MID$(Q$,I,1)=" " AND V$=
"" THEN V$=LEFT$(Q$,I-1)
```

Questa linea cerca in Q$ uno spazio bianco. Se ne trova uno, e V$ è ancora vuota, scrive in V$ tutte le lettere a sinistra dello spazio.




```
170 IF MID$(Q$,I+1,1)<>" " AND
V$<>"" THEN W$=MID$(Q$,I+1,
LEN(Q$)-1):I=LEN(Q$)
```

Continua a cercare in Q$ finché non trova una lettera dopo uno spazio. Prende quindi tutto quel che si trova a destra dello spazio e lo scrive in W$ (questo vuol dire che il numero di spazi che il giocatore pone tra le parole non ha nessuna importanza).

Quando V$ e W$ sono entrambe riempite, il contatore è posto al valore massimo, per uscire dal ciclo.

```
180 NEXT I
190 IF W$="" THEN V$=Q$
```

Se il computer non ha trovato spazi tra le lettere, allora V$ e W$ saranno ancora vuote alla fine del ciclo. Allora il computer prende l'intero contenuto di Q$ e lo scrive in V$.

```
200 M$="QUESTE SONO LE TUE DUE PAROLE"
210 PRINT "PRIMA PAROLA=";V$
220 PRINT "SECONDA PAROLA=";W$
230 IF W$="" THEN M$="HAI BATTUTO
UNA SOLA PAROLA"
240 IF W$="" AND V$="" THEN M$=
"NON HAI BATTUTO NULLA"
250 PRINT M$
260 STOP
```

Questa sezione serve a far girare da sola la routine che divide le parole. Vengono quindi stampati messaggi in funzione di quello che avete inserito. Fatela girare per vedere cosa succede.

L'analisi dell'inserimento

Il computer ora ha in memoria le istruzioni del giocatore contenute in due stringhe di caratteri V$ (stringa dei verbi) e W$ (stringa di parole). Compito successivo sarà di controllare tali istruzioni confrontandole con le parole che gli avete dato nella procedura di inizializzazione. Il computer presuppone che la parola contenuta in V$ sia un verbo, e compie una ricerca per cicli successivi, per far corrispondere tale parola con uno dei verbi contenuti nella matrice V$( ). (Notate la differenza tra la variabile stringa V$ e la matrice V$( ) per il computer sono due cose diverse: cercate di non confonderle)!..

Il computer quindi procede allo stesso modo per corrispondere il contenuto di W$ con una delle parole contenute nella matrice O$( ). Ecco la sezione di programma che controlla la corrispondenza tra le parole inserite e le parole nella memoria del computer.

```
10 VB=0
```

VB è una nuova variabile predisposta per contenere il numero della casella del verbo corrispondente. Es: se il giocatore ha battuto il verbo "PIGLIARE", il computer cerca nella matrice V$( ) finché non raggiunge la casella 10, dove è contenuto "PIGLIARE". In questo caso, VB = 10.




```
20 FOR I=1 TO V
30 IF V$=V$(I) THEN VB=I
40 NEXT I
```

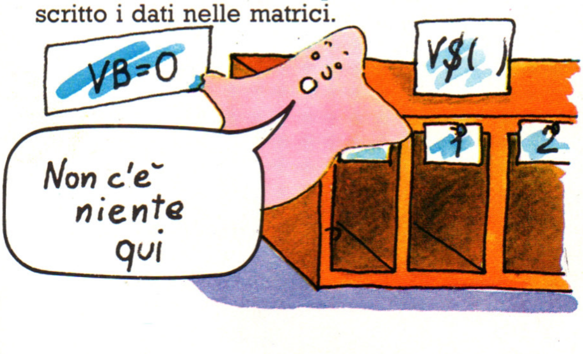
Il computer cicla per V volte (V = n° verbi in memoria) confrontando il verbo dato dal giocatore con quelli in memoria. Se trova quello che corrisponde, attribuisce a VB il valore opportuno.

```
50 OB=0
60 FOR I=1 TO W
70 IF W$=O$(I) THEN OB=I
80 NEXT I
```

Il ciclo per W$ funziona allo stesso modo, utilizzando OB per registrare il numero di casella della parola corrispondente a quella data.

Se le parole non corrispondono?

Nel caso non vi siano corrispondenze, VB e OB saranno entrambe uguali a 0. Il computer interpreta ciò come se la parola corrispondente a quella fornita si trovasse nella casella zero. Ma quando va a cercare in tale casella la parola corrispondente, non vi trova nulla, perché l'avevate lasciata vuota quando avevate scritto i dati nelle matrici.

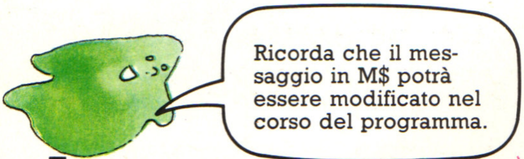


Combinazioni prive di senso

Attenzione, questo processo di ricerca della corrispondenza si limita a controllare che le due parole date siano in memoria, ma non controlla che la combinazione delle due parole abbia senso. Una combinazione priva di senso, come: DISSERRARE LA CANDELA, viene accettata in questa fase del programma, ma verrà rifiutata successivamente quando il computer proverà ad eseguire l'azione richiesta. A questo punto del programma è molto piú rapido limitarsi a controllare le parole separatamente, piuttosto che dire al computer di controllare la validità delle combinazioni. Al termine di questa sezione del programma il computer ha un valore per VB ed uno per OB.

Predisporre i messaggi d'errore

Il computer può usare i valori di VB e OB per controllare se deve mandare al giocatore un messaggio di istruzioni non valide. Questa parte del programma funziona da filtro. Le istruzioni fornite dal giocatore sono sottoposte ad alcuni test. Se non superano anche uno solo dei test, viene posto un nuovo messaggio in M$. Se invece superano tutti i test, M$ conterrà ancora il messaggio "COSA?", scritto inizialmente alla linea 220. (In questa fase i messaggi sono predisposti in memoria, ma non vengono stampati sullo schermo, e possono essere cambiati, nel corso del programma). Ecco le linee di programma che predispongono i messaggi di errore della "Casa Stregata" cercate di individuarli nel listato principale. Se scrivete da soli un'avventura dovrete includere anche linee di questo tipo.



1

```
IF W$>"" AND OB=0 THEN M$="CHE
SCIOCCHEZZA"
```



Il primo test controlla che vi sia una parola in W$ (ovvero che il giocatore abbia inserito due parole) e quindi controlla se il valore di OB sia nullo.

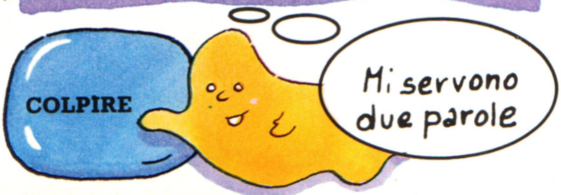
2

```
IF VB=0 THEN VB=V+1
```

Questa linea serve a superare un problema. Non potete usare l'istruzione GOSUB con valore zero. I diversi tipi di BASIC variano nel tipo di risposta. La maggior parte di essi ignora il GOSUB e continua con la linea successiva. Altri invece fanno obiezioni allo zero, ed emettono un messaggio di "errore di intervallo". Per superare questa difficoltà, VB viene cambiato da zero ad un valore maggiore di V (il numero di verbi nella memoria del computer) e il computer viene inviato ad una subroutine fittizia.

3

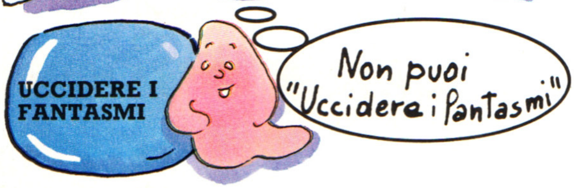
```
IF W$="" THEN M$="MI SERVONO
DUE PAROLE"
```



Questa linea prepara un messaggio nel caso in cui il giocatore abbia inserito solo una parola, e W$ sia vuota. (Se la parola fa parte di quei comandi di una sola parola permessi al giocatore, questo messaggio verrà modificato piú avanti).

4

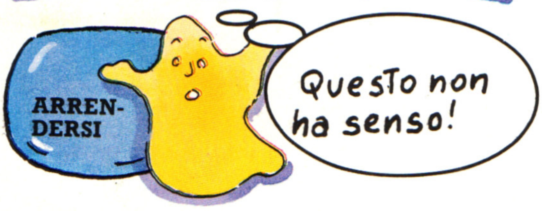
```
IF VB>V AND OB>0 THEN M$="NON
PUOI '"+Q$+"'"
```



Questa linea dispone un messaggio se il computer non trova il verbo dato fra quelli in memoria, ma trova il nome dell'oggetto.

5

```
IF VB>V AND OB=0 THEN M$="QUESTO
NON HA SENSO"
```



Se il computer non trova in memoria nessuna delle due parole fornite dal giocatore, predispone questo messaggio.

NOVITA' ASSOLUTA IN EDICOLA

Guida VIDEO GIOCHI

1
GIUGNO
L. 3.500

LA GRANDE GUIDA A TUTTI I GIOCHI ELETTRONICI E NON

Nuovissima, ricca e tutta a colori. GUIDA VIDEOGIOCHI ti aspetta in edicola con oltre 60 giochi recensiti, i commenti, le curiosità, i trucchi e le novità da tutto il mondo.

E, in più, partecipi al grande concorso riservato ai fedeli lettori di GUIDA VIDEOGIOCHI.

FANTASTICO CONCORSO GUIDA VIDEOGIOCHI

I premi in palio sono favolosi: due esclusive Control Deck NINTENDO e tanti game originali.

Nintendo

MATTEL



GRUPPO EDITORIALE JACKSON

Aut. Min. Rich.

CPC464 e 6128 fantastici computer, fantastici TV!

L. 399.000 + IVA



TUTTO COMPRESO.

CPC464GT 64 Kb RAM con monitor fosfori verdi, tastiera, registratore a cassetta, joystick, 100 programmi/giochi: L. 399.000.+ IVA

CPC464CTM 64 Kb RAM con monitor a colori, tastiera, registratore a cassette, joystick, 100 programmi/giochi: L. 699.000.+ IVA

CPC6128GT 128 Kb RAM con monitor a fosfori verdi, velocissimo disk driver da 3" doppia faccia (180 Kb + 180 Kb), joystick, 50 programmi/giochi: L. 699.000.+ IVA

CPC6128CTM 128 Kb RAM con monitor a colori, velocissimo disk driver da 3" doppia faccia (180 Kb + 180 Kb), joystick, 50 programmi/giochi: L. 899.000.+ IVA

WKS 6128 TV.

Stazione completa composta da: CPC 6128 CTM; Tavolo a ripiani; Sintonizzatore TV; Antenna amplificata.
Tutto a L. 999.000.+ IVA

PRONTO AMSTRAD.

Telefonaci: 02/26410511, avrai ogni informazione; oppure scrivici: Casella Postale 10794 - 20124 Milano.

LI TROVI QUI.

Presso i numerosissimi punti vendita Amstrad. Cerca quello più vicino su "Amstrad Magazine" in edicola, chiedi anche Junior Amstrad la rivita che ti regala i giochi per CPC (troverai molte notizie in più).
Oltre 150 Centri di Assistenza Tecnia.

FANTASTICO, DIVENTA TV COLOR.

Al momento del tuo acquisto puoi trasformare il tuo CPC con monitor a colori in TV color, il tuo TV color, come?

Ma è semplice, basta Acquistare il sintonizzatore TV (MP3) a L. 199.000.+ IVA



AMSTRAD

DALLA PARTE DEL CONSUMATORE

OPINIONE Comunicazione Integrata